



UNIVERSIDAD
POLITÉCNICA
DE MADRID

PROCESO DE
COORDINACIÓN DE LAS
ENSEÑANZAS PR/CL/001



E.T.S. de Ing. de Sistemas
Informáticos

ANX-PR/CL/001-01

GUÍA DE APRENDIZAJE

ASIGNATURA

615000255 - Evolucion Y Mantenimiento Del Software

PLAN DE ESTUDIOS

61IW - Grado En Ingenieria Del Software

CURSO ACADÉMICO Y SEMESTRE

2025/26 - Segundo semestre

Índice

Guía de Aprendizaje

1. Datos descriptivos.....	1
2. Profesorado.....	1
3. Conocimientos previos recomendados.....	2
4. Competencias y resultados de aprendizaje.....	2
5. Descripción de la asignatura y temario.....	3
6. Cronograma.....	6
7. Actividades y criterios de evaluación.....	9
8. Recursos didácticos.....	12
9. Otra información.....	13

1. Datos descriptivos

1.1. Datos de la asignatura

Nombre de la asignatura	615000255 - Evolucion y Mantenimiento del Software
No de créditos	6 ECTS
Carácter	Obligatoria
Curso	Tercero curso
Semestre	Sexto semestre
Período de impartición	Febrero-Junio
Idioma de impartición	Castellano
Titulación	61IW - Grado en Ingeniería del Software
Centro responsable de la titulación	61 - E.T.S De Ing. De Sistemas Informáticos
Curso académico	2025-26

2. Profesorado

2.1. Profesorado implicado en la docencia

Nombre	Despacho	Correo electrónico	Horario de tutorías *
Jessica Diaz Fernandez	1119	yesica.diaz@upm.es	Sin horario.
Juan Manuel Garitagoitia Martin	1101	juanmanuel.garitagoitia@upm.es	Sin horario.
Angel Panizo Lledot (Coordinador/a)	1214	angel.panizo@upm.es	Sin horario.

Carolina Gallardo Perez		carolina.gallardop@upm.es	Sin horario.
-------------------------	--	---------------------------	--------------

* Las horas de tutoría son orientativas y pueden sufrir modificaciones. Se deberá confirmar los horarios de tutorías con el profesorado.

3. Conocimientos previos recomendados

3.1. Asignaturas previas que se recomienda haber cursado

- Bases De Datos
- Ingeniería De Requisitos Y Modelado
- Fundamentos De Ingeniería Del Software
- Ingeniería Del Proceso Software Y Construcción

3.2. Otros conocimientos previos recomendados para cursar la asignatura

- Conocimientos básicos de programación en .NET (ASP.NET)
- Conocimientos básicos de programación en Java

4. Competencias y resultados de aprendizaje

4.1. Competencias

CC5 - Conocimiento, administración y mantenimiento de sistemas, servicios y aplicaciones informáticas.

CE1 - Capacidad para desarrollar, mantener y evaluar servicios y sistemas software que satisfagan todos los requisitos del usuario y se comporten de forma fiable y eficiente, sean asequibles de desarrollar y mantener y cumplan normas de calidad, aplicando las teorías, principios, métodos prácticos de la Ingeniería del Software.

CT5 - Organización y planificación: Identificar y definir eficazmente las metas, objetivos y prioridades de una tarea o proyecto a desempeñar estipulando las actividades, los plazos y los recursos requeridos y controlando los procesos establecidos.

4.2. Resultados del aprendizaje

RA82 - Identifica y define eficazmente las metas, objetivos y prioridades de una tarea o proyecto a desempeñar estipulando las actividades, los plazos y los recursos requeridos y controlando los procesos establecidos

RA69 - Utiliza entornos y herramientas de desarrollo

RA105 - Identifica y analiza problemas para solventar soluciones software sobre la base de un conocimiento adecuado de teorías, modelos y técnicas actuales.

RA98 - Mantiene soluciones atendiendo a la evolución del software

RA107 - Desarrolla, evalúa y mantiene sistemas software que satisfacen requisitos de usuario

5. Descripción de la asignatura y temario

5.1. Descripción de la asignatura

La asignatura tiene como objetivo global dotar al alumno de los conocimientos fundamentales, teóricos y prácticos, de los procesos de mantenimiento y evolución del software.

La asignatura tiene como objetivos específicos que el alumno sea capaz de poner en práctica patrones de mantenibilidad del software así como procesos de gestión de la configuración empleando técnicas y herramientas avanzadas para la gestión del cambio, control de versiones e integración continua. El alumno aplicará técnicas de reingeniería e ingeniería inversa para recuperar sistemas legados.

5.2. Temario de la asignatura

1. Mantenimiento y Evolución del Software

1.1. Proceso de Mantenimiento y Evolución

1.1.1. Introducción al mantenimiento y evolución del software

1.1.2. El cambio y las leyes del cambio

1.1.3. Tipos de mantenimiento

1.1.4. Modelos de proceso software. Proceso de Evolución. Estándares ? ISO 14764

1.2. Proceso de Gestión de la Configuración del Software

1.2.1. Gestión del cambio

1.2.2. Control de Versiones

1.2.3. Gestión de la compilación, despliegue y entrega

1.3. Desarrollo, mantenimiento y operación continuos ? DevOps

1.3.1. Integración Continua

1.3.2. Entrega Continua

1.3.3. Despliegue Continuo

2. Control de Versiones

2.1. Gestión de la configuración y Control de Versiones

2.2. Tipos de Sistemas de Control de Versiones

2.3. Sistema de Control de Versiones Git y Github

3. Mantenibilidad del Software

3.1. Mantenibilidad & Refactorización

3.2. Bad smells y Análisis de Código Estático

3.3. Principios de mantenibilidad

3.4. Deuda técnica (Technical Debt)

4. Reingeniería. Sistemas legados

4.1. Reingeniería. Ingeniería inversa e Ingeniería directa

4.2. Ingeniería Inversa - enfoque práctico

4.3. Reingeniería de sistemas legados: Wrapping & Migración

4.4. Estrategias de migración

6. Cronograma

6.1. Cronograma de la asignatura *

Sem	Actividad tipo 1	Actividad tipo 2	Tele-enseñanza	Actividades de evaluación
1	Clase Teoría de Presentación Duración: 02:00 LM: Actividad del tipo Lección Magistral Clase Teoría Tema 1 Duración: 02:00 LM: Actividad del tipo Lección Magistral			
2	Clase Teoría Tema 1 Duración: 02:00 LM: Actividad del tipo Lección Magistral	Trabajo 1.1. Estándar Proceso de Mantenimiento Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
3		Trabajo 1.2. Integración continua Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio Trabajo 1.3. Integración y despliegue continuo Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
4		Trabajo 1.3. Integración y despliegue continuo Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio Trabajo 1.4. Integración y despliegue continuo Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		
5		Trabajo 1.4. Integración y despliegue continuo Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio Trabajo 1.4. Integración y despliegue continuo Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio		

6		<p>Trabajo 1.5. Integración y despliegue continuo Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio</p> <p>Trabajo 1 Resumen y cuestionario Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio</p>		
7				
8	<p>Clase Teoría Tema 2 Duración: 02:00 LM: Actividad del tipo Lección Magistral</p> <p>Clase Teoría Tema 2 Duración: 02:00 LM: Actividad del tipo Lección Magistral</p>			
9		<p>Trabajo 2. Uso de Git/Github Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio</p> <p>Trabajo 2. Uso de Git/Github Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio</p>		
10	<p>Clase Teoría Tema 3 Duración: 02:00 LM: Actividad del tipo Lección Magistral</p> <p>Clase Teoría Tema 3 Duración: 02:00 LM: Actividad del tipo Lección Magistral</p>			
11		<p>Trabajo 3. Patrones de mantenibilidad Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio</p> <p>Trabajo 3. Patrones de mantenibilidad Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio</p>		
12		<p>Trabajo 3. Patrones de mantenibilidad Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio</p> <p>Trabajo 3. Patrones de mantenibilidad Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio</p>		<p>[Eval-Progresiva] Tems 1-2-3 & Trabajos 1-2-3 [NO RECUPERABLE] OT: Otras técnicas evaluativas Evaluación Progresiva y Global Presencial Duración: 00:00</p>
13	<p>Clase Teoría Tema 4 Duración: 02:00 LM: Actividad del tipo Lección Magistral</p>	<p>Trabajo4: Reingeniería de un sistema legado Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio</p>		

14		<p>Trabajo4: Reingeniería de un sistema legado Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio</p> <p>Trabajo4: Reingeniería de un sistema legado Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio</p>		
15		<p>Trabajo4: Reingeniería de un sistema legado Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio</p> <p>Trabajo4: Reingeniería de un sistema legado Duración: 02:00 PL: Actividad del tipo Prácticas de Laboratorio</p>		<p>[Eval-Progresiva] Trabajo 4 [NO RECUPERABLE] PG: Técnica del tipo Presentación en Grupo Evaluación Progresiva y Global Presencial Duración: 00:00</p>
16				
17				<p>[Eval-Global] Examen EX: Técnica del tipo Examen Escrito Evaluación Progresiva y Global Presencial Duración: 02:00</p>

Para el cálculo de los valores totales, se estima que por cada crédito ECTS el alumno dedicará dependiendo del plan de estudios, entre 26 y 27 horas de trabajo presencial y no presencial.

7. Actividades y criterios de evaluación

7.1. Actividades de evaluación de la asignatura

7.1.1. Evaluación (progresiva)

Sem.	Descripción	Modalidad	Tipo	Duración	Peso en la nota	Nota mínima	Competencias evaluadas
12	[Eval-Progresiva] Temas 1-2-3 & Trabajos 1-2-3 [NO RECUPERABLE]	OT: Otras técnicas evaluativas	Presencial	00:00	15%	0 / 10	CT5 CC5 CE1
15	[Eval-Progresiva] Trabajo 4 [NO RECUPERABLE]	PG: Técnica del tipo Presentación en Grupo	Presencial	00:00	15%	0 / 10	CC5 CE1 CT5
17	[Eval-Global] Examen	EX: Técnica del tipo Examen Escrito	Presencial	02:00	70%	5 / 10	CC5 CE1 CT5

7.1.2. Prueba evaluación global

Sem	Descripción	Modalidad	Tipo	Duración	Peso en la nota	Nota mínima	Competencias evaluadas
12	[Eval-Progresiva] Temas 1-2-3 & Trabajos 1-2-3 [NO RECUPERABLE]	OT: Otras técnicas evaluativas	Presencial	00:00	15%	0 / 10	CT5 CC5 CE1
15	[Eval-Progresiva] Trabajo 4 [NO RECUPERABLE]	PG: Técnica del tipo Presentación en Grupo	Presencial	00:00	15%	0 / 10	CC5 CE1 CT5
17	[Eval-Global] Examen	EX: Técnica del tipo Examen Escrito	Presencial	02:00	70%	5 / 10	CC5 CE1 CT5

7.1.3. Evaluación convocatoria extraordinaria

Descripción	Modalidad	Tipo	Duración	Peso en la nota	Nota mínima	Competencias evaluadas
[Eval-Extra] Examen	EX: Técnica del tipo Examen Escrito	Presencial	02:00	100%	5 / 10	CC5 CE1 CT5

7.2. Criterios de evaluación

EVALUACIÓN PROGRESIVA

Actividad de Evaluación	Resultado de Aprendizaje
Temas 1-2-3 & Trabajos 1-2-3	RA105 Identifica y analiza problemas para solventar soluciones software sobre la base de un conocimiento adecuado de las teorías, modelos y técnicas actuales RA69 Utiliza entornos y herramientas de desarrollo RA98 Mantiene soluciones atendiendo a la evolución del software RA107 Desarrolla, evalúa y mantiene sistemas software que satisfacen requisitos de usuario
Trabajo 4	RA105 Identifica y analiza problemas para solventar soluciones software sobre la base de un conocimiento adecuado de las teorías, modelos y técnicas actuales RA69 Utiliza entornos y herramientas de desarrollo RA98 Mantiene soluciones atendiendo a la evolución del software RA107 Desarrolla, evalúa y mantiene sistemas software que satisfacen requisitos de usuario RA82 - Identifica y define eficazmente las metas, objetivos y prioridades de una tarea o proyecto a desempeñar estipulando las actividades, los plazos y los recursos requeridos y controlando los procesos establecidos.

EVALUACIÓN GLOBAL

Actividad de Evaluación	Resultado de Aprendizaje
Examen	<p>RA98 Mantiene soluciones atendiendo a la evolución del software</p> <p>RA69 Utiliza entornos y herramientas de desarrollo</p> <p>RA107 Desarrolla, evalúa y mantiene sistemas software que satisfacen requisitos de usuario</p> <p>RA82 - Identifica y define eficazmente las metas, objetivos y prioridades de una tarea o proyecto a desempeñar estipulando las actividades, los plazos y los recursos requeridos y controlando los procesos establecidos.</p> <p>RA105 Identifica y analiza problemas para solventar soluciones software sobre la base de un conocimiento adecuado de las teorías, modelos y técnicas actuales.</p>

El cálculo de Nota Final (NF)= 0.7 x Examen + 0.3 x Evaluación de las pruebas de progresiva

Para superar la asignatura se debe obtener una calificación mayor o igual a 5 sobre 10.

Durante el curso se realizarán evaluaciones en vivo en las sesiones de prácticas, pueden ser en forma de presentaciones grupales o defensas orales frente al profesor. Dichas evaluaciones tienen un peso del 30% y dada su naturaleza no se pueden recuperar en la convocatoria ordinaria.

CONVOCATORIA EXTRAORDINARIA

Para el adelanto de la convocatoria extraordinaria según NORMATIVA DE EVALUACIÓN DEL APRENDIZAJE EN LAS TITULACIONES OFICIALES DE GRADO Y MÁSTER UNIVERSITARIO DE LA UNIVERSIDAD POLITÉCNICA DE MADRID (Aprobada por Consejo de Gobierno en su sesión del 26 de mayo de 2022) no se establece ningún requisito específico.

Debido a la naturaleza oral de las "Evaluación de prácticas realizada durante clase" estas no se tendrán en cuenta en la convocatoria extraordinaria. Un alumno que suspenda la asignatura en la convocatoria ordinaria perderá la

nota que tuviese en las "Evaluación progresivas" para la convocatoria extraordinaria.

* Las soluciones de los exámenes cuando sean de tipo test no se publicarán dado el grado limitado de variabilidad.

* * La competencia transversal CT5 (Organización y Planificación) se evalúa mediante las pruebas de tipo Examen Escrito en las que el alumno ha tenido que organizar y planificar sus laboratorios para adquirir los conocimientos necesarios para superar dichas pruebas.

8. Recursos didácticos

8.1. Recursos didácticos de la asignatura

Nombre	Tipo	Observaciones
Software Engineering 9 Edition	Bibliografía	Software Engineering 9 Edition., Sommerville
Software Evolution and Maintenance, a practitioner?s approach	Bibliografía	Software Evolution and Maintenance, a practitioners approach, Priyadarshi Tripathy, Kshirasagar Naik. Wiley 2015
Building maintainable software, C# Edition	Bibliografía	Building maintainable software, C# Edition, by Joost Visser, Sylvan Rigal, Gijs Wijnholds, Pascal van Eck, Rob van der Leek. O'Reilly Media 2016
Building Maintainable Software, Java Edition	Bibliografía	Building Maintainable Software, Java Edition, by Joost Visser, Sylvan Rigal, Rob van der Leek, Pascal van Eck, Gijs Wijnholds. O'Reilly Media 2016
Refactoring: Improving the Design of Existing Code	Bibliografía	Refactoring: Improving the Design of Existing Code, by Martin Fowler and Kent Beck ADDISON WESLEY 2008
Pro Git	Bibliografía	https://progit2.s3.amazonaws.com/es/2016-03-23-ea7f6/progit-es.1091.pdf

GitHub Glossary	Recursos web	https://help.github.com/articles/github-glossary/
Git Reference	Recursos web	https://git-scm.com/docs http://gitref.org/
Equipamiento aula	Equipamiento	
Equipamiento Prácticas	Equipamiento	

9. Otra información

9.1. Otra información sobre la asignatura